Network Working Group
Request For Comments #327
NIC #9261

Abhay Bhushan
MIT-MAC
April 27, 1972

Rcvd 19 May 72

Data and File Transfer Workshop Notes

On April 14 and 15, 1972, a Data and File Transfer Workshop was held at M.I.T., Cambridge, Mass. A list of attendees of the meeting for April 14 and 15 is appended to the notes. This note attempts to summarize most of the topics discussed and all of the decisions reached at the workshop meeting.

The following is a summary of the talks and discussions on April 14, 1972.

Steve Crocker discussed a general theory for Network protocols. Protocols transformations should have a unique inverse, and should be transitive. Transformation to a standard form requires only 2n transformations (n = number of different types of hosts), as compared with n(n-1) transformations with no standard form. A standard approach is preferable for $n \geq 3$.

For file transfer, one could define a Network Virtual File Image. There was some discussion on whether it was possible to satisfy the above rules for file structure transformations. No agreement was reached and the problem was abandoned for the present.

Further discussion lead to the following formulation of the Workshop goals:

To come up with data and file transfer protocol/strategy that satisfies the needs of ARPANET users including Maxi-HOSTs, Mini-HOSTs, TIPs, Datacomputer, RJE, and Mailbox users.

Goals for the protocols/strategy were set as:

1. It should preserve the integrity of data.
2. It should preserve the integrity of character representation and interpretation.
3. It should preserve the integrity of structural information, to the extent conveniently possible.
4. It should lead to the development of a Network Virtual File System.

Richard Winter discussed the Datacomputer application. The Datacomputer though usable from terminals directly will not be engineered for direct terminal users, but for use by programs. In Datalanguage a user can define data and file structure, and also how the file/data is to be transferred. Using the data language it is possible to transfer entire files, or only the relevant parts of files. The following is an example of file transfer as currently envisioned in the Datacomputer.

LOGIN <user> <password>
CREATE <file name> <description>
CREATE <port name> <description>
PORT <port name> <external name>
<file name> = <port name> (for transfer to Datacomputer)
<port name> = <file name> (for transfer from Datacomputer)
LOGOUT

(CREATE statements are needed only when the description(s) required are not already on file at the Datacomputer. A port description can specify a standard "external name", thus making a

port statement optional also. "External name" is to be a HOST-socket specification. The data transfer is to be in accordance with network data transfer standards. The File and Port descriptions are to be in Datalanguage.)

Alex McKenzie discussed the TIP user needs, describing the current capabilities and limitations of TIPs and TIP terminals. TELNET format is the first choice of TIP users, followed by DTP using the indefinite bit stream mode. There are two TIPs with magnetic tape systems which are capable of transferring data between them using the current DTP (RFC 264) in the descriptor count mode (utilizing sequence number option).

Bob Braden discussed the RJS protocol and presented some data on RJS use. NETRJS is 1% of CCN job load representing 2,000 jobs, 10,000 sessions and 1,000 hours connect time in the last 5 months. Average job input is of the order of 100,000 bits (400 cards), average job output is 700,000 bits (1,000 lines). Large files have about 10 million bits representing about 8-10 minutes of transmission time. The RJS protocol will be defined in a forthcoming document.

Ray Tomlinson described the CPYNET system BBN is using to transfer files among TENEX systems. CPYNET commands are ASCII strings with a fixed syntax. The original connection is closed after a command is accepted, and data is transferred on a new connection using previous socket number, but with possibly a different byte size. The data transfer rate achieved in CPYNET has been about 10 Kb/s.

Abhay Bhushan discussed the evaluation of network protocols and presented some preliminary measurement results. The evaluation criteria for protocols should include speed (real time delay and transmission rate), efficiency (cpu time or cost), reliability (error rate and failure rate), convenience (ease of use and implementation), and usage (suitability for various application and user classes).

The parameters that affect speed and efficiency for given system conditions (fixed load, etc.) are:

1) Byte size used for NCP connection.
2) Average message size used for transmission.
3) Data format conversion (e.g., into Network ASCII, DTP Blocks, etc.)
4) Buffer size and I/O mode used (unit or block mode, etc.).
5) Other protocol constraints (acknowledge, error checking, connection procedure, etc.).

There was some discussion as to how data and file transfer protocols may be altered to make transfer faster and more efficient by using optimum byte size and minimizing some of the constraints that impose a large overhead.

The follow up discussions on DTP and FTP lead to a list of discussion and decision items for the next day. The following is a summary of decisions reached on Saturday, April 15, 1972.

1. Separate connections are to be used for data and control information.
2. Control connection is to be a "TELNET" full-duplex

connection (NVT-ASCII), established via the ICP. Data connections are to be simplex connections established directly.

3. The File Transfer and File System commands and their arguments shall be printable ASCII strings, instead of numeric codes, so that they are directly usable by a user at a terminal. The interaction, however, will be optimized for usage by programs (indirect use).
4. The byte size and user socket for data connection, data representation, and transfer mode to be used in file transfer may be chosen by a user via one or more commands requiring a positive or negative acknowledgment.
5. The following data representations are to be accepted by all servers:

   1) Network ASCII (7-bit ASCII in 8-bit field with 8th bit zero).

   2) Local Byte (a server option to store data in an efficient manner, the storage scheme should be well publicized).

   3) Image (a sequence of bits which should be stored contiguously independent of the byte size chosen for transfer).

   4) ASCII Print file (convert ASCII file to a form suitable for printing).

   5) EBCDIC Print file (convert EBCDIC file to a form suitable for printing).

6. Record structures are allowed but not mandatory. A user with no record structure in his file should be able to store or retrieve his file at any host. If a serving host cannot accept record structure, it must inform the

user of the fact. Any record structure information in the data stream may subsequently be discarded.

7. The following data transfer modes are defined:

   1) Byte-Stream - End of File indicated by closing connection. No record structure.

   2) Block - File is series of blocks preceded by a count field. Appropriate means provided to indicate end-of-file, end-of-record, and restart markers.

   3) ASCII - The file is network-ASCII, end-of-record, and end-of-file are indicated by a special "TELNET-control" character with 8th bit set to "one".

   4) File is network-ASCII with an end-of-record defined by CR LF, and end-of-file by closing connection.

   5) Hasp compressed file with end-of-record and end-of-file information.

8. A restart procedure will be provided to protect user from system failures (either host or process dying). The issue of bits lost or scrambled is handled best at the NCP level. Standard error codes and responses will be provided for storage and I/O channel errors, at the FTP level.

   The restart procedure would require that the sender of data insert a special marker in the data stream (the marker has meaning only to the sender. It could be bit-count, record count, or page count, etc.). The receiver of data would mark the corresponding position of this marker in its own system, and return this information to the user. In the event of a system failure, the user can restart the transfer by supplying this information with a restart command.

9. DTP is no longer a separate protocol but a set of transfer modes or format procedures whose use is defined by the file transfer protocol.

10. Abhay Bhushan will write the workshop notes and the draft specifications for the new file transfer protocol.

List of Attendees, Data and File Transfer Workshop

Abhay Bhushan	MIT-MAC	April 14, 15
Bob Braden	UCLA-CCN	April 14, 15
Arvola Chan	MIT-MAC	April 14, 15
Steve Crocker	ARPA	April 14
Eric Harslem	RAND	April 14
John Heafner	RAND	April 14
Chuck Holland	UCSD	April 14, 15
Alex McKenzie	BBN (NET)	April 14
Bob Metcalfe	MIT-MAC	April 14
Hal Murray	CCA	April 14
Bill Plummer	BBN	April 14
Jon Postel	UCLA	April 14
Neal Ryan	MIT-MAC	April 14, 15
Marc Seriff	MIT-MAC	April 14, 15
Bob Thomas	BBN	April 14
Ray Tomlinson	BBN	April 14
Dick Watson	SRI-ARC	April 14, 15
Doug Wells	MIT-MAC	April 14
Jim White	SRI-ARC	April 14, 15
Richard Winter	CCA	April 14, 15

9. DTP is no longer a separate protocol but a set of
transfer modes or format procedures whose use is defined
by the file transfer protocol.

10. Abbey Bhushan will write the workshop notes and the draft
specifications for the new file transfer protocol.

List of Attendees, Date and File Transfer Workshop

Abbey Bhushan	MIT-MAC	April 14, 15
Bob Braden	UCLA-CCN	April 14, 15
Arpple Cohen	MIT-MAC	April 14, 15
Steve Crocker	ARPA	April 14
Eric Harslem	RAND	April 14
John Heafner	RAND	April 14
Chuck Hoffman	UCSD	April 14, 15
Alex Mckenzie	BBN (NET)	April 14
Bob Metcalfe	MIT-MAC	April 14
Mel Merrey	CCA	April 14
Bill Plummer	BBN	April 14
Jon Postel	UCLA	April 14
Sean Ryan	MIT-MAC	April 14, 15
Marc Seriff	MIT-MAC	April 14, 15
Bob Thomas	BBN	April 14
Ray Tomlinson	BBN	April 14
Dick Watson	SRI-ARC	April 14, 15
Doug Wells	MIT-MAC	April 14
Jim White	SRI-ARC	April 14, 15
Richard Winter	CCA	April 14, 15